CLAIMS

What is claimed is:

1. A structure for representing a query statement having an atomic query element and a combined query element related by a combined operator comprising comprising:

a superclass, further comprising:

a first subclass representing the atomic query element;

a second subclass representing the combined query element, wherein each of the left and right subelements can be any subclass of the superclass; and

a relationship indicator representing a relationship between the first subclass and the second subclass as defined by the combined operator.

2. The structure of claim 1, wherein at least one class of the superclass further comprises zero or more types for the query element represented by that class and a subclass defined for each identified type.

3. The structure of claim 1, wherein at least one subclass further comprises a superclass.

4. The structure of claim 1, wherein the superclass represents a table reference, the first subclass represents an unjoined table and the second subclass represents a joined table.

5. The structure of claim 1, wherein the superclass represents a value expression, the first subclass represents an atomic value expression and the second subclass comprises a combined value expression.

6. The structure of claim 1, wherein the superclass represents a search condition, the first subclass represents an atomic search condition, and the second subclass represents a combined search condition.

7. The structure of claim 1, wherein the superclass represents a group-by query element, the first subclass represents a group, and the second subclass represents a grouping set.

8. The structure of claim 1, wherein the second subclass further comprises a nested query language element.

9. The structure of claim 1, wherein the second subclass represents an iterative query language element.

10. The structure of claim 1, further comprising means for receiving a query statement having an atomic query element and a combined query element associated by a combined operator; and means for populating the structure with the received query statement.

11. The structure of claim 10, further comprising means for receiving the query statement from a user-interface.

12. The structure of claim 10, further comprising means for receiving the query statement from an application interface.

13. The structure of claim 3, further comprising:

means responsive to selection of a class or subclass instance of the populated model, for retrieving only the query elements populating the selected class or subclass instance and all subclasses of that class or subclass instance; and

means for building a query statement from the retrieved query elements using the relationships defined by the hierarchical class structure of the model.

14. The structure of claim 2, further comprising:

means for identifying a first query element type for a first query language dialect;

means for identifying at least a second query element type for at least a second query language dialect, the second element type being substantially functionally equivalent to the first query element type; and

means for creating a generic subclass representative of both the identified first and at least second element type.

15. A method for hierarchically representing a query statement having an atomic query element and a combined query element related by a combined operator comprising the steps of:

defining a superclass representing the query element;

defining a first subclass of the superclass representing the atomic query element;

defining a second subclass of the superclass representing the combined query element, wherein each of the left and right subelements comprises any class of the superclass; and

indicating a relationship between the first subclass and the second subclass defined by the combined operator.

16. The method of claim 15, further comprising the step of:

for at least one class of the superclass, identifying zero or more types for the query element represented by that class and defining a subclass for each identified type.

17. The method of claim 15, wherein at least one subclass further comprises a superclass.

18. The method of claim 15, wherein the superclass represents a table reference, the first class represents an unjoined table and the second class represents a joined table.

19. The method of claim 15, wherein the superclass represents a value expression, the first class represents an atomic value expression and the second class comprises a combined value expression.

20. The method of claim 15, wherein the superclass represents a search condition, the first class represents an atomic search condition, and the second class represents a combined search condition.

21. The method of claim 15, wherein the superclass represents a group-by query element, the first class represents a group, and the second class represents a grouping set.

22. The method of claim 15, wherein the second class further comprises a nested query language element.

23. The method of claim 15, wherein the second class represents an iterative query language element.

24. The method of claim 15, further comprising the steps of:

receiving a query statement having an atomic query element and a combined query element associated by a combined operator; and

populating the structure with the received query statement.

25. The method of claim 15, further comprising the step of receiving the query statement from a user-interface.

26. The method of claim 15, further comprising the step of receiving the query statement from an application interface.

27. The method of claim 17, further comprising the steps of:

in response to a selection of a class or subclass instance of the populated model, retrieving only the query elements populating the selected class or subclass instance and all subclasses of that class or subclass instance; and

building a query statement from the retrieved query elements using the relationships defined by the hierarchical class structure of the model.

28. The method of claim 16, further comprising the steps of:

i) identifying a first query element type for a first query language dialect;

ii) identifying at least a second query element type for at least a second query language dialect, the second element type being substantially functionally equivalent to the first element type; and

iii) creating a subclass representative of both the identified first and at least second element types.

29. An article of manufacture comprising a computer program carrier readable by a computer and embodying one or more instructions executable by the computer for providing a structure for representing a query statement having an atomic query element and a combined query element related by a combined operator, the computer program comprising:

program instructions defining a superclass, further comprising:

program instructions for defining a first subclass representing the atomic query element;

program instructions for defining a second subclass representing the combined query element, wherein each of the left and right subelements can be any subclass of the superclass; and

program instructions for defining a relationship indicator representing a relationship between the first subclass and the second subclass as defined by the combined operator.

30. The article of manufacture of claim 29, wherein at least one class of the superclass further comprises zero or more types for the query element represented by that class and a subclass defined for each identified type.

31. The article of manufacture of claim 29, wherein at least one subclass further comprises a superclass.

32. The article of manufacture of claim 29, wherein the superclass represents a table reference, the first subclass represents an unjoined table and the second subclass represents a joined table.

33. The article of manufacture of claim 29, wherein the superclass represents a value expression, the first subclass represents an atomic value expression and the second subclass comprises a combined value expression.

34. The article of manufacture of claim 29, wherein the superclass represents a search condition, the first subclass represents an atomic search condition, and the second subclass represents a combined search condition.

35. The article of manufacture of claim 29, wherein the superclass represents a group-by query element, the first subclass represents a group, and the second subclass represents a grouping set.

36. The article of manufacture of claim 29, wherein the second subclass further comprises a nested query language element.

37. The article of manufacture of claim 29, wherein the second subclass represents an iterative query language element.

38. The article of manufacture of claim 29, further comprising program instructions for receiving a query statement having an atomic query element and a combined query element associated by a combined operator; and program instructions for populating the structure with the received query statement.

39. The article of manufacture of claim 38, further comprising program instructions for receiving the query statement from a user-interface.

40. The article of manufacture of claim 38, further comprising programinstructions for receiving the query statement from an application interface.

41. The article of manufacture of claim 31, further comprising:

program instructions responsive to selection of a class or subclass instance of the populated model, for retrieving only the query elements populating the selected class or subclass instance and all subclasses of that class or subclass instance; and

program instructions for building a query statement from the retrieved query elements using the relationships defined by the hierarchical class structure of the model.

42. The article of manufacture of claim 30, further comprising:

program instructions for identifying a first query element type for a first query language dialect;

program instructions for identifying at least a second query element type for at least a second query language dialect, the second element type being substantially functionally equivalent to the first query element type; and program instructions for creating a generic subclass representative of both the identified first and at least second element type.